

AMENDMENTS TO THE CLAIMS

Please cancel claims 4 and 14 without prejudice.

Please amend the claims as follows:

1. (Currently amended) A method comprising:
receiving an application program;
compiling the application program into a first compiled version for execution by a
first processor, compiling the application including instrumenting the first
compiled version with monitoring instructions to direct the capture of
profile data;
executing the first compiled version using the first processor;
capturing profile data during the execution of the first compiled version as
directed by the monitoring instructions of the first compiled version; and
compiling the application program into a second compiled version for execution
by a second processor, the compiling of the second compiled version
including optimization based at least in part on the captured profile data.
2. (Original) The method of claim 1, further comprising storing the profile data in a
memory.
3. (Original) The method of claim 1, further comprising executing the second
compiled version using the second processor.
4. (Cancelled)

5. (Original) The method of claim 1, wherein the second processor is an embedded processor.
6. (Original) The method of claim 5, wherein the second processor is not capable of capturing profile data.
7. (Original) The method of claim 5, wherein the second processor is not capable of generating external communications.
8. (Original) The method of claim 1, wherein the first processor is a host processor for a device and wherein the device includes the second processor.
9. (Original) The method of claim 1, wherein compiling the application program into a first compiled version utilizes a first compiler and wherein compiling the application program into a second compiled version utilizes a second compiler.
10. (Original) The method of claim 1, wherein compiling the application program into a first compiled version and compiling the application program into a second compiled version are performed with a single compiler.
11. (Currently amended) A machine-readable medium having stored thereon data representing instructions that, when executed by a processor, cause the processor to perform operations comprising:
receiving an application program;
compiling the application program into a first compiled version for execution by a
first processor, compiling the application including instrumenting the first

compiled version with monitoring instructions to direct the capture of profile data;

executing the first compiled version using the first processor;

capturing profile data during the execution of the first compiled version as

directed by the monitoring instructions of the first compiled version; and

compiling the application program into a second compiled version for execution

by a second processor, the compiling of the second compiled version

including optimization based at least in part on the captured profile data.

12. (Original) The medium of claim 11, wherein the instructions include instructions that, when executed by a processor, cause the processor to perform operations comprising storing the profile data in a memory.
13. (Original) The medium of claim 11, wherein the instructions include instructions that, when executed by a processor, cause the processor to perform operations comprising executing the second compiled version using the second processor.
14. (Cancelled)
15. (Original) The medium of claim 11, wherein the second processor is an embedded processor.
16. (Original) The medium of claim 15, wherein the second processor is not capable of capturing profile data.
17. (Original) The medium of claim 15, wherein the second processor is not capable of generating external communications.

18. (Original) The medium of claim 11, wherein the first processor is a host processor for a device and wherein the device includes the second processor.
19. (Original) The medium of claim 11, wherein compiling the application program into a first compiled version utilizes a first compiler and wherein compiling the application program into a second compiled version utilizes a second compiler.
20. (Original) The medium of claim 11, wherein compiling the application program into a first compiled version and compiling the application program into a second compiled version are performed with a single compiler.
21. (Currently amended) A system comprising:
one or more memories, data being stored within the one or memories including a first compiler and a second compiler, the first compiler compiling an application program into a first compiled version, the first compiled version including monitoring instruction instrumentation to direct the capture of profile data;
a host microprocessor, the host microprocessor executing the first compiled version, the host microprocessor capturing profile data during the execution of the first compiled version according to the monitoring instruction instrumentation of the first compiled version; and
a target processor, the second compiler compiling the application code into a second compiled version for execution by the target processor, the second compiled version being optimized based at least in part on the captured profile data.

22. (Original) The system of claim 21, wherein the captured profile data is stored in the one or more memories.
23. (Original) The system of claim 21, wherein the target microprocessor is an embedded microprocessor.
24. (Original) The system of claim 23, wherein the target microprocessor does not have the capability of capturing a profile data.
25. (Original) The system of claim 23, wherein the target microprocessor does not have the capability of generating external communications.
26. (Currently amended) A method of optimizing the execution of a program by an embedded processor comprising:
obtaining the program;
compiling the program to generate a first set of compiled code, the first set of compiled code being instrumented with instructions to monitor the execution of the first set of compiled code;
executing the first set of compiled code on a host processor, the host processor being contained in a device that also contains the embedded processor;
capturing profile information during the execution of the first set of compiled code according to the instrumented instructions of the first set of compiled code and saving the profile information in a memory;
compiling the program to generate a second set of compiled code, the second set of compiled code being optimized based at least in part on the captured profile information; and

executing the second set of compiled code using the embedded processor.

27. (Original) The method of claim 26, wherein the first set of compiled code is compiled utilizing a first compiler and the second set of compiled code is compiled utilizing a second compiler.
28. (Original) The method of claim 26, wherein the first set of compiled code and the second set of compiled code are compiled utilizing a single compiler.